

# Macro Layout

Layout is our invisible conductor—a silent rhythm that brings structure to interaction, movement to content, and clarity to every experience. In Play+, layout is not just a frame—it is the stage where adaptability meets intention.

Rooted in our Adaptive and Intuitive pillars, our layout system responds fluidly, guides effortlessly, and feels exactly right—across screens, flows, and user states. Our principles of rhythm, hierarchy, and harmony reduce cognitive overload and preserve focus. Layout breathes life into our Fluid soul and fuels our Metamorphic ambitions—interfaces that intelligently reflow and adapt without losing coherence.

Good layout should not be noticed. It should be felt .

## Our Core Layout Principles

Our layout system is governed by five key principles:

Principle	Design Guideline	Developer Implementation
Hierarchical	Establish a clear visual hierarchy; prominence reflects importance.	Use semantic HTML ( <code>&lt;main&gt;</code> , <code>&lt;nav&gt;</code> , <code>&lt;aside&gt;</code> ) and structure-defining components.
Consistent	Apply a single spacing/grid system everywhere; eliminate visual guesswork.	Use spacing tokens ( <code>\$space-*</code> ), not magic numbers. Adhere to grid columns.
Adaptable	Design layouts that fluidly reflow across devices and contexts.	Use mobile-first, relative units ( <code>rem</code> , <code>%</code> ). Apply min-width media queries.

Principle	Design Guideline	Developer Implementation
Efficient	Prioritize scan-ability and intent clarity. Avoid clutter and ambiguity.	Group related items, use white space to create breathing room and guide the eye.
Inclusive	Ensure navigable, logical, and accessible layouts for everyone.	Preserve logical DOM order, visible focus states, and avoid layout shifts on interaction.

## The Spacing System (8pt Grid)

We use an 8pt spacing system to create rhythm, balance, and harmony. All spacing, padding, and margins are multiples of 8px, ensuring a uniform cadence across devices and layouts.

Token	Value (px)	Use Case
`\$space-xxs`	4	Fine-tuning within tight UI (e.g., icon+text, badges).
`\$space-xs`	8	Spacing between closely related elements (e.g., chips, tags).
`\$space-sm`	16	Internal padding for small containers (e.g., cards, buttons).
`\$space-md`	24	Gaps between medium components, inter-card spacing.
`\$space-lg`	32	Section spacing, larger paddings.
`\$space-xl`	48	Major group separation (e.g., sidebar vs content).

Token	Value (px)	Use Case
`\$space-xxl`	64	Hero spacing, visual anchoring for key moments.

## The 12-Column Grid System

Play+ uses a 12-column responsive grid to structure content across screens. It ensures adaptability without rigidity, and flexibility without chaos.

Breakpoint	Screen Range	Columns	Gutter	Description
Extra Small (xs)	0 - 599px	4	16px	Compact stacking for handhelds
Small (sm)	600 - 904px	8	24px	Tablets, split views
Medium (md)	905 - 1239px	12	24px	Default web layout size
Large (lg)	1240 - 1439px	12	32px	Desktops, dashboard spaces
Extra Large (xl)	1440px+	12	32px	High-res workspace, full-grid use

## Structural Patterns & Best Practices

### Common Page Layouts

### Persistent Elements

### Scrolling Best Practices

Type	When to Use
Vertical	Default behavior. Keep it unified and page-wide.
Inner/Nested	Use rarely, with clear affordances. Acceptable for dropdowns, tables, panes.
Horizontal	Avoid unless displaying uncollapsible tabular data. Provide scroll cues.

## Future State: AI-Powered Adaptive Layouts

We envision a future where layout decisions are made dynamically—powered by user context, behavior, and goals.

### Conceptual Use Cases:

Play+ is laying the groundwork today—using consistent structure—so that future interfaces can breathe and evolve naturally.

### Guardrails